

UNITED STATES PATENT APPLICATION

OF

Young Jun JUNG

For

INTER-PROCESSOR COMMUNICATION METHOD AND APPARATUS

FOR MOBILE COMMUNICATION SYSTEM

1003450-1394
TOP SECRET

[0001] This application claims the benefit of the Korean Application No. P2000-86587 filed on December 30, 2000, which is hereby incorporated by reference.

BACKGROUND OF THE INVENTION

Field of the Invention

[0002] The present invention relates to a mobile communication system, and more particularly, to inter-processor communication method/apparatus for a communication system, in which a message is transmitted between the call-processing and maintenance processors without any connection management in an upper user layer.

Discussion of the Related Art

[0003] In general, a Transmission Control Protocol/Internet Protocol (TCP/IP) includes a network access layer, an internet layer, a transport layer, and an application layer. The transport layer includes a TCP that provides connection-oriented services and a User Datagram Protocol (UDP) that provides connectionless services.

[0004] The TCP that provides the connection-oriented services is responsible for managing data transmission processes between

terminals, and it provides a data communication environment for communication entities of its upper layers. On the other hand, the UDP providing the connectionless services simply transmits each data packet to a corresponding destination address.

[0005] The TCP/IP layers being able to provide connection-oriented services are not embodied in the base stations and control stations of an existing mobile communication system. Therefore, a message gets transmitted using a packet-based message processing technique in a connectionless manner. Then, the network side performs a message routing process. For this reason, the existing mobile communication system uses a nonstandard protocol.

[0006] As described above, since each communication system uses an internal nonstandard protocol, it lacks the data transmission compatibility between data processors included in the system. In addition, the data handling processes related to the connectionless technique involve an additional data process for preventing any data loss, reducing the efficiency rate of the whole communication system.

SUMMARY OF THE INVENTION

[0007] Accordingly, the present invention is directed to inter-processor communication method and apparatus for a mobile

communication system that substantially obviates one or more problems due to limitations and disadvantages of the related art.

[0008] An object of the present invention is to provide inter-processor communication method and apparatus for a mobile communication system that enables a connectionless-oriented user to communicate with a connection-oriented data transmission system.

[0009] Additional advantages, objects, and features of the invention will be set forth in part in the description which follows and in part will become apparent to those having ordinary skill in the art upon examination of the following or may be learned from practice of the invention. The objectives and other advantages of the invention may be realized and attained by the structure particularly pointed out in the written description and claims hereof as well as the appended drawings.

[0010] To achieve these objects and other advantages and in accordance with the purpose of the invention, as embodied and broadly described herein, an inter-processor communication method for a mobile communication system includes (a) receiving a message transmission request from a user, the request including a message and a destination address of the message; (b) determining whether any one of currently existing sockets, whose file descriptors are stored in a socket management database, is connected to the destination address; and (c) sending a

connection request to a connection manager in order to be connected to a TCP (transmission control protocol) layer, if it is determined that none of the existing sockets are connected to the destination address.

[0011] The method further includes creating a new socket connected to the destination address and attempting to be connected to the TCP layer; storing a new file descriptor of the new socket in the database if the attempt is succeeded; and of newly forming a receiving module for the new socket.

[0012] In another aspect of the present invention, an inter-processor communication apparatus for a mobile communication system includes a socket management database storing file descriptors of currently existing sockets; a message-transmitting module receiving a message transmission request from a user, the request including a message and a destination address of the message, the module further sending a connection request in order to be connected to a TCP (transmission control protocol) layer if none of the existing sockets are connected to the destination address; and a connection manager creating a new socket connected to the destination address and attempting to connect the module with the TCP layer after receiving the connection request from the module.

[0013] It is to be understood that both the foregoing general description and the following detailed description of the present

invention are exemplary and explanatory and are intended to provide further explanation of the invention as claimed.

BRIEF DESCRIPTION OF THE DRAWINGS

[0014] The accompanying drawings, which are included to provide a further understanding of the invention and are incorporated in and constitute a part of this application, illustrate embodiment(s) of the invention and together with the description serve to explain the principle of the invention. In the drawings;

[0015] FIG. 1 illustrates the interface layers of a communication system according to the present invention; and

[0016] FIG. 2 illustrates the functional blocks of the TCP message router according to the present invention.

DETAILED DESCRIPTION OF THE INVENTION

[0017] Reference will now be made in detail to the preferred embodiments of the present invention, examples of which are illustrated in the accompanying drawings. Wherever possible, the same reference numbers will be used throughout the drawings to refer to the same or like parts.

[0018] The Transmission Control Protocol (TCP) message router according to the present invention includes a network equipment for routing a standard TCP/IP packet, an operating system (OS)

that includes a standard TCP/IP protocol, a TCP/IP connection manager, and message transmitting and receiving modules.

[0019] The network equipment can be any one of an Asynchronous Transfer Mode (ATM) switch, an ATM user connecting system, a fast Ethernet hub, and workstations/embedded systems/PCs that include a fast Ethernet hub.

[0020] FIG. 1 illustrates the interface layers of a communication system according to the present invention. As shown in FIG. 1, a TCP message router receives a stream-based message from a connectionless-oriented user and sends the message to a socket interface through a socket.

[0021] Then the socket interface (I/F) sends the message to a TCP or User Datagram Protocol (UDP) layer. The TCP or UDP layer formats the message into a protocol data unit (PDU) and sends the formatted PDU to a network through an Ethernet layer or the ATM based IP (IPOA) layer.

[0022] The method of processing the TCP/IP based messages in accordance with the present invention includes the steps of connecting a connectionless-oriented user to a connection-oriented TCP, transmitting the messages, and receiving messages. References to these steps will now be made in detail using FIG. 2, which illustrates the functional blocks of the TCP message router according to the present invention.

[0023] First of all, the message router establishes the connection between a connectionless-oriented user and a connection-oriented TCP interface. The TCP message router does not attempt to connect to the TCP interface unless it receives a message transmission request from a user.

[0024] When the TCP message router receives a message transmission request from a user, a transmitting module determines whether any one of currently existing sockets, whose file descriptors (FD) are stored in a socket management database, is connected to the destination address.

[0025] If there is, the transmitting module sends the message to a TCP layer. Otherwise, it makes a connection request to a connection manager included in the TCP message router and waits to receive a reply.

[0026] When the connection manager receives a connection request from the transmitting module, it creates a new socket and attempts to connect to the TCP layer. If the connection manager succeeds in connecting to the TCP layer, it adds the socket information of the new socket to the socket management database and newly installs a receiving module corresponding to the new socket.

[0027] However, if it fails to connect to the TCP layer, the connection manager waits until it receives a connection request from another user. In this case, the socket gets deleted

automatically. If the connection manager fails to connect to the TCP layer in a given time, the transmitting module informs the user of the failure.

[0028] When the connection manager receives another connection request from outside, it checks whether there is a socket that uses the address of the communication object that made the request. If there is, the connection manager deletes the socket and creates a new socket.

[0029] In addition, when a connection cancellation request or error is made in any socket whose FD is included in the database, it automatically closes the socket and eliminates the socket FD from the database. At this time, the connection manager may open a UDP socket for packet transmission through a UDP.

[0030] The Inter-Processor Communication (IPC) Daemon shown in FIG. 2 is a functional element that automatically runs a background program for inter-processor communication, and the TCP link keep-alive block checks whether the TCP is linked properly. The IPC Daemon and TCP link block can update the socket management database alternately. However, more than two functional blocks may refer to the database at same time.

[0031] Second of all, the transmitting module performs the message transmission process. When the transmitting module receives a message transmission request from a user, the module determines whether any one of currently existing sockets, whose

file descriptors are stored in a socket management database, is connected to the destination address. If there is, it sends the message to a TCP layer through the socket. Otherwise, it makes a connection request to the connection manager and waits for a reply. If the transmitting module fails to connect to the TCP layer in a given time, it informs the user of the failure.

[0032] A stream-based message handled by the TCP may not be transmitted at once, and there is no indication of the beginning of the message. Therefore, the TCP adds a message header that indicates the beginning of the message to the message. The message header includes a message header indicator, a message length, a source address, a destination address, and a message Identifier (ID).

[0033] The transmitting module makes a transmission request by invoking a Service Access Function provided by the TCP. After the message is sent, the transmitting module receives the size of a portion of the message, which is actually sent. Subsequently, it sends the remaining portion of the message.

[0034] If the transmitting module does not complete the message transmission in a given time, it reports the failure. In addition, in a case where an error occurs in the Service Access Function, the transmitting module closes the socket, and reports the error.

[0035] When the transmitting module wants to send a message through a particular UDP socket, it initially searches for the socket. If the socket does not exist, the transmitting module makes a connection request and transmits the data.

[0036] Finally, a receiving module receives a message sent by a user. Because the TCP handles the stream-based data, a message can be received in the form of one or more data subsets, or more than one messages can be received at once.

[0037] When the receiving module receives the data subsets of a message, it stores the subsets in a circular queue in an order they were received. Thereafter, the data subsets are converted to a single data packet, and the packet is transmitted to a corresponding user.

[0038] A receiving module gets formed when a socket is formed, and it gets deleted when the socket is closed.

[0039] Namely, if the receiving module reads the byte stream of a message in a socket, it stores the message in a circular queue. Thereafter, it searches for a message indicator. If it finds the message indicator, it checks whether the data stored in the circular queue correspond to the message size indicated in the message header. If it still needs to receive more data, it waits until any data is read in the socket.

[0040] After checking the message header, it checks whether the data body included in the header is read. If the reading is

completed, the receiving module sends the message to a destination user through an internal IPC. Otherwise, it waits until the data reading is completed. By repeating the steps described above, the entire message can be transmitted to the destination user.

[0041] A receiving module gets formed for each socket. The receiving module receives a message through a socket and transmits the message to the user.

[0042] As described above, a mobile communication system that handles packet-based data is able to use a TCP/IP that handles stream-based data by using the method and apparatus of the present invention. Therefore, an independent message processing mechanism can be provided for each final user. The present invention may also be applied to an IMT-2000 system or all other IP systems.

[0043] It will be apparent to those skilled in the art that various modifications and variations can be made in the present invention. Thus, it is intended that the present invention covers the modifications and variations of this invention provided they come within the scope of the appended claims and their equivalents.